BDD, User Stories, Intro Diagramming, and moving into Rails

2/9/2016
Announcements

• HW0 graded- regrades due by Thursday
• HW2 will be out tonight (due 2/22)
• Project, Iteration 0-2 due 2/15
  – Meet with groups at end of class
    • Group 1: Developing for Team 2
    • Group 2: Developing for Team 4
      – Group 3: Developing for Team 8
      – Group 4: Developing for Team 9
    – Group 5: Developing for Team 6
    – Group 6: Developing for Team 7
    – Group 7: Developing for Team 5
    – Group 8: Developing for Team 3
    – Group 9: Developing for Team 1
User Stories

- 1-3 sentences in everyday language
  - Fits on 3” x 5” index card
  - Written by/with customer

- “Connextra” format:
  - Feature name
  - As a [kind of stakeholder],
  - So that [I can achieve some goal],
  - I want to [do some task]
  - 3 phrases must be there, can be in any order

- Idea: user story can be formulated as acceptance test before code is written
Why 3x5 Cards?

• (from User Interface community)
• Nonthreatening => all stakeholders participate in brainstorming
• Easy to rearrange => all stakeholders participate in prioritization
• Since stories must be short, easy to change during development
  – As often get new insights during development
Different stakeholders may describe behavior differently

• *See which of my friends are going to a show*
  – As a theatergoer
  – So that I can enjoy the show with my friends
  – I want to see which of my Facebook friends are attending a given show

• *Show patron’s Facebook friends*
  – As a box office manager
  – So that I can induce a patron to buy a ticket
  – I want to show her which of her Facebook friends are going to a given show
Product Backlog

- Real systems have 100s of user stories
- **Backlog**: User Stories not yet completed
  - (We’ll see Backlog again with Pivotal Tracker)
- Prioritize so most valuable items highest
- Organize so they match SW releases over time
SMART User Stories
SMART stories

- **S**pecific
- **M**easurable
- **A**chievable
  (ideally, implement in 1 iteration)
- **R**elevant
  (“the 5 why’s”)
- **T**imeboxed
  (know when to give up)
Specific & Measurable

• Each scenario testable
  – Implies known good input and expected results exist

• Anti-example:
  “UI should be user-friendly”

• Example: Given/When/Then.
  – Given some specific starting condition(s),
  – When I do X,
  – Then one or more specific thing(s) should happen
Achievable

• Complete in 1 iteration
• If can’t deliver feature in 1 iteration, deliver subset of stories
  – Always aim for working code @ end of iteration
Timeboxed

• Estimate what’s achievable using velocity
  – Each story assigned points (1-3) based on difficulty
  – Velocity
    = Points completed / iteration
  – Use measured velocity to plan future iterations & adjust points per story

• Pivotal Tracker (later) tracks velocity
Relevant: “business value”

• Ask “Why?” recursively until discover business value, or kill the story:
  – Protect revenue
  – Increase revenue
  – Manage cost
  – Increase brand value
  – Making the product remarkable
  – Providing more value to your customers
Stories are SMART—but features should be relevant

• Specific & Measurable: can I test it?
• Achievable? / Timeboxed?
• Relevant? use the “5 whys”
• Show patron’s Facebook friends

As a box office manager
So that I can induce a patron to buy a ticket
I want to show her which Facebook friends are going to a given show
Lo-Fi UI Sketches and Storyboards
SaaS User Interface Design

• SaaS apps often faces users
  \( \Rightarrow \) User stories need User Interface (UI)
• Want *all* stakeholders involved in UI design
  – Don’t want UI rejected!
• Need UI equivalent of 3x5 cards
• **Sketches**: pen and paper drawings or “Lo-Fi UI”
Lo-Fi UI Example

(Figure 4.3, *Engineering Long Lasting Software* by Armando Fox and David Patterson, Alpha edition, 2012.)
Storyboards

- Need to show how UI changes based on user actions
- HCI => “storyboards”
- Like scenes in a movie
- But not linear
Example Storyboard

(Figure 4.4, *Engineering Long Lasting Software* by Armando Fox and David Patterson, Alpha edition, 2012.)
Lo-Fi to HTML

• Tedious to do sketches and storyboards, but easier than producing HTML!
  – Also less intimidating to nontechnical stakeholders
    => More likely to suggest changes to UI if not code behind it
  – More likely to be happy with ultimate UI

• Next steps: More on CSS (Cascading Style Sheets) and Haml
  – Make it pretty after it works